

WEST[Help](#)[Logout](#)[Interrupt](#)[Main Menu](#)[Search Form](#)[Posting Counts](#)[Show S Numbers](#)[Edit S Numbers](#)[Preferences](#)**Search Results -**

Terms	Documents
(object-oriented OR (object ADJ oriented))AND (717/???.ccls. OR 717/?.ccls.) AND (workflow OR work ADJ flow) AND RULES	0

Database:
 US Patents Full-Text Database
 US Pre-Grant Publication Full-Text Database
 JPO Abstracts Database
 EPO Abstracts Database
 Derwent World Patents Index
 IBM Technical Disclosure Bulletins

Refine Search:
 (object-oriented OR (object ADJ oriented))AND (717/???.ccls. OR 717/?.ccls.) AND (workflow OR work ADJ flow) AND RULES
 [Clear](#)

Search History**Today's Date: 6/13/2001**

<u>DB Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u>
TDBD	(object-oriented OR (object ADJ oriented))AND (717/???.ccls. OR 717/?.ccls.) AND (workflow OR work ADJ flow) AND RULES	0	L5
DWPI	(object-oriented OR (object ADJ oriented))AND (717/???.ccls. OR 717/?.ccls.) AND (workflow OR work ADJ flow) AND RULES	0	L4
EPAB	(object-oriented OR (object ADJ oriented))AND (717/???.ccls. OR 717/?.ccls.) AND (workflow OR work ADJ flow) AND RULES	0	L3
JPAB	(object-oriented OR (object ADJ oriented))AND (717/???.ccls. OR 717/?.ccls.) AND (workflow OR work ADJ flow) AND RULES	0	L2
USPT	(object-oriented OR (object ADJ oriented))AND (717/???.ccls. OR 717/?.ccls.) AND (workflow OR work ADJ flow) AND RULES	8	L1

WEST[Help](#)[Logout](#)[Interrupt](#)[Main Menu](#)[Search Form](#)[Posting Counts](#)[Show 8 Numbers](#)[Edit 8 Numbers](#)[Preferences](#)**Search Results -**

Terms	Documents
(object-oriented OR (object ADJ oriented))AND (717/?? .ccls. OR 717/? .ccls.) AND (workflow OR work ADJ flow) AND RULES	8

Database:

US Patents Full-Text Database	▲
US Pre-Grant Publication Full-Text Database	
JPO Abstracts Database	
EPO Abstracts Database	
Derwent World Patents Index	
IBM Technical Disclosure Bulletins	▼

Refine Search:	<table><tr><td>(object-oriented OR (object ADJ oriented))AND (717/?? .ccls. OR 717/? .ccls.) AND (workflow OR work ADJ</td><td>▲</td></tr><tr><td></td><td>■</td></tr><tr><td></td><td>▼</td></tr></table>	(object-oriented OR (object ADJ oriented))AND (717/?? .ccls. OR 717/? .ccls.) AND (workflow OR work ADJ	▲		■		▼	Clear
(object-oriented OR (object ADJ oriented))AND (717/?? .ccls. OR 717/? .ccls.) AND (workflow OR work ADJ	▲							
	■							
	▼							

Search History**Today's Date: 6/13/2001**

<u>DB Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u>
USPT	(object-oriented OR (object ADJ oriented))AND (717/?? .ccls. OR 717/? .ccls.) AND (workflow OR work ADJ flow) AND RULES	8	<u>L1</u>

WEST[Generate Collection](#)**Search Results - Record(s) 1 through 8 of 8 returned.**☐ 1. Document ID: US 6170081 B1

L1: Entry 1 of 8

File: USPT

Jan 2, 2001

US-PAT-NO: 6170081

DOCUMENT-IDENTIFIER: US 6170081 B1

TITLE: Method and system for interfacing to a variety of software development tools

DATE-ISSUED: January 2, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Fontana; James Albert	Mission Viejo	CA	N/A	N/A
Pitchford; Anthony Reginald	Mission Viejo	CA	N/A	N/A
Smith; Christopher Eyre	Coto de Caza	CA	N/A	N/A
Tadman; Mark Jeffrey	Mission Viejo	CA	N/A	N/A

US-CL-CURRENT: 717/1; 707/10, 707/103Z, 707/513, 709/213, 709/246, 709/332, 717/10

ABSTRACT:

A method and system for facilitating use of a tool in heterogeneous environments and application categories in a software development framework having a storage device. First, a context object is created for storing all intermediate information generated while the tool is being used. Next, the specific environment in which the tool is going to be used is identified information about the environment is stored in the context object. The specific tasks the tool typically performs are identified and searched for any previously accomplished tasks in the framework. The results of the search are stored in the context object. Information needed for the tool to operate is retrieved from the repository and the information is supplied as input files to the tool. The tool is run with the input files and the output derived is stored as a result of running the tool. The context object is updated by analyzing the output derived from the tool. Then, the analyzed output of the operation performed by the tool is stored in the repository for the environment.

14 Claims, 5 Drawing figures Exemplary Claim Number: 1
Number of Drawing Sheets: 5

Full	Title	Citation	Front	Review	Classification	Date	Reference	Claims	KMMC	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	--------	------	-----------	-------

☐ 2. Document ID: US 6167564 A

L1: Entry 2 of 8

File: USPT

Dec 26, 2000

US-PAT-NO: 6167564

DOCUMENT-IDENTIFIER: US 6167564 A

TITLE: Software system development framework

DATE-ISSUED: December 26, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Fontana; James Albert	Mission Viejo	CA	N/A	N/A
Iyengar; Sridhar Srinivasa	Irvine	CA	N/A	N/A
Pitchford; Anthony Reginald	Mission Viejo	CA	N/A	N/A
Smith; Norman Roy	Lake Forest	CA	N/A	N/A
Tolbert; Douglas Marshall	Newport Beach	CA	N/A	N/A

US-CL-CURRENT: 717/1; 717/11, 717/2

ABSTRACT:

A system and method in a computer system for integrating software development tools and applications into the computer system in order to build, deploy and maintain enterprise business process applications in a heterogeneous development framework. Integration of the applications and software development tools are achieved through integration of the key elements of the computer system which are business models, domain models and components. In the process of integration the origin of a first newly developed/modified/existing business model is traced to a first newly developed/modified/existing domain model and these models are linked together. Next, the constituent components of a second newly developed/modified/existing domain model are traced to a newly developed/modified/existing set of components created and linked together. The system also involves recovery of constituent components from a newly developed/modified/existing system in a first heterogeneous environment and those constituent components are reconstructed into usable components inside a third newly developed/modified/existing domain model and are linked together. The process also involves recovery of a fourth newly developed/modified/existing domain model from a second heterogeneous environment and linking it to a second newly developed/modified/existing business model.

1 Claims, 14 Drawing figures Exemplary Claim Number: 1
Number of Drawing Sheets: 14

Full	Title	Citation	Front	Review	Classification	Date	Reference	Claims	KUMC	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	--------	------	-----------	-------

☐ 3. Document ID: US 6158044 A

L1: Entry 3 of 8

File: USPT

Dec 5, 2000

US-PAT-NO: 6158044
DOCUMENT-IDENTIFIER: US 6158044 A

TITLE: Proposal based architecture system

DATE-ISSUED: December 5, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Tibbetts; John J.	San Francisco	CA	N/A	N/A

US-CL-CURRENT: 717/1; 707/1, 709/310

ABSTRACT:

A proposal based architecture system that converts a transaction submission process into a generic object in a computer environment. A preferred embodiment of the invention provides a tool set which allows the user to create a set of Proposal Specifications which define the structure of the possible components of a Proposal. The user defines the complete characteristics (meta-data) for any kind of Proposal which define the hierarchy of domain relationships, interaction modes, validation references, and assumptions. The actual Proposal instance is formed using the definitions in the Proposal Specifications. A Proposal allows a user to add, change, and annotate data, is self aware and navigates between pages and skips to appropriate fields automatically and supports n-level undo/redo. It also tracks all versions of data updates and the user that is responsible for each data update, recognizes and corrects stale data, and enables long-lived transactions, off-line transaction processing, and collaborative transactions. A Proposal is accessed via multiple User Interfaces (UI), breaking the close coupling between the front-end and the back-end and allowing the user to add a Web or Graphical User Interface (GUI) front-end without having to rewrite the back-end application. A UI coordinator maps user input fields to components of the Proposal and communicates with different user interfaces such as: Internet; Graphical User Interface (GUI); Object Oriented User Interface (OOUI); proprietary interface; and devices such as bar code readers or keypads. The invention also provides a default UI and components. A set of Transaction Processing/Data Processing (TP/DP) interfaces are provided to communicate with back-end transactional interfaces such as Database Management Systems (DBMS), Transactional Processing (TP) Monitors, and Object Oriented Databases.

75 Claims, 30 Drawing figures Exemplary Claim Number: 1
Number of Drawing Sheets: 19

Full	Title	Citation	Front	Review	Classification	Date	Reference	Claims	KWIC	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	--------	------	-----------	-------

☐ 4. Document ID: US 6074431 A

L1: Entry 4 of 8

File: USPT

Jun 13, 2000

US-PAT-NO: 6074431
DOCUMENT-IDENTIFIER: US 6074431 A

TITLE: Apparatus and method for automatically forming work object and recording medium recording work object forming program and capable of being read by computer

DATE-ISSUED: June 13, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Watanabe; Jun	Kawasaki	N/A	N/A	JPX
Hosozawa; Haruko	Kawasaki	N/A	N/A	JPX

US-CL-CURRENT: 717/2

ABSTRACT:

A work process is divided into data and behavior of the data and is arranged. Each data is prepared as a business object. As for the behavior of the data, a necessary parts list table (processing description file) is formed by analyzing a user construction control pattern formed by obtaining answers regarding work rules about how to handle the data by the user on the basis of a question table. A customized work object is formed from the necessary parts list table and the business object (class definition file).

19 Claims, 41 Drawing figures Exemplary Claim Number: 1
Number of Drawing Sheets: 31

Full	Title	Citation	Front	Review	Classification	Date	Reference	Claims	RMC	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	--------	-----	-----------	-------

☐ 5. Document ID: US 5960200 A

L1: Entry 5 of 8

File: USPT

Sep 28, 1999

US-PAT-NO: 5960200
DOCUMENT-IDENTIFIER: US 5960200 A

TITLE: System to transition an enterprise to a distributed infrastructure

DATE-ISSUED: September 28, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Eager; Timothy	Fullerton	CA	N/A	N/A
Anand; Madhav	Cambridge	MA	N/A	N/A
Aslanian; Edouard	Hermosa Beach	CA	N/A	N/A

US-CL-CURRENT: 717/5; 703/13, 703/20, 705/7, 709/201, 717/1, 717/7

ABSTRACT:

An automated system transitions an entire enterprise to a distributed infrastructure. The system includes a process for organizing and managing the transition, a multi-tiered client/server architecture that adheres to open systems standards, a system to automate the transition of existing applications to this architecture, and a system to enable the creation or modification of applications based on this architecture.

54 Claims, 36 Drawing figures Exemplary Claim Number: 1
Number of Drawing Sheets: 36

Full	Title	Citation	Front	Review	Classification	Date	Reference	Claims	KMIC	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	--------	------	-----------	-------

☐ 6. Document ID: US 5930512 A

L1: Entry 6 of 8

File: USPT

Jul 27, 1999

US-PAT-NO: 5930512

DOCUMENT-IDENTIFIER: US 5930512 A

TITLE: Method and apparatus for building and running workflow process models using a hypertext markup language

DATE-ISSUED: July 27, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Boden; Edward Barnes	Vestal	NY	N/A	N/A
Hansen; Robert Christian	Johnson City	NY	N/A	N/A
Leska; Michael Anthony	Endicott	NY	N/A	N/A
Paxhia; Frank Vincent	Binghamton	NY	N/A	N/A
Sylvester; Scott Anthony	Endicott	NY	N/A	N/A

US-CL-CURRENT: 717/10

ABSTRACT:

Apparatus and computer implemented method for process modeling using both a web server and a workflow server in process definition. During buildtime, a process definition in the language of some workflow process modeler is fed to transform $f(p)$, which provides a translate process definition into a hypertext markup language (HTML) representation of the workflow process. A user utilizing the facilities of an HTTP server may operate a web browser to request that HTTP server provide the HTML process definition to enable the user to create, or read and modify, the process definition as desired. The result is then fed through an inverse transform $f'(p)$ to a workflow server in the language of that server, for invocation and execution during runtime of the programs defined during buildtime.

29 Claims, 20 Drawing figures Exemplary Claim Number: 3

Number of Drawing Sheets: 19

Full	Title	Citation	Front	Review	Classification	Date	Reference	Claims	KMIC	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	--------	------	-----------	-------

☐ 7. Document ID: US 5923879 A

L1: Entry 7 of 8

File: USPT

Jul 13, 1999

US-PAT-NO: 5923879
DOCUMENT-IDENTIFIER: US 5923879 A

TITLE: Conversion system and method between corba and c/c++ architectures for
corba data pairs/couples

DATE-ISSUED: July 13, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Sasmazel; Levent M.	Holmdel	NJ	N/A	N/A
Schneider; David H.	Manalapan	NJ	N/A	N/A

US-CL-CURRENT: 717/5; 707/102, 709/313

ABSTRACT:

A conversion system converts data between CORBA and C/C++ environments. The conversion system includes a format definition database storing first information including, for example, user definitions, C/C++ specific information and/or CORBA specific information. A system dependent database stores second information including, for example, operating system and processor specific information. A parser parses the first and second information into a data dictionary. A decoder generates C/C++ memory structures using the data dictionary and CORBA data couples. An encoder generates CORBA data couples using the data dictionary and C/C++ memory structures.

17 Claims, 8 Drawing figures Exemplary Claim Number: 1
Number of Drawing Sheets: 7

Full	Title	Citation	Front	Review	Classification	Date	Reference	Claims	KOMC	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	--------	------	-----------	-------

☐ 8. Document ID: US 5920725 A

L1: Entry 8 of 8

File: USPT

Jul 6, 1999

US-PAT-NO: 5920725

DOCUMENT-IDENTIFIER: US 5920725 A

TITLE: Run-time object-synthesis and transparent client/server updating of distributed objects using a meta server of all object descriptors

DATE-ISSUED: July 6, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Ma; Henry Chi-To	Redwood City	CA	N/A	N/A
Lo; George C.	Fremont	CA	N/A	N/A

US-CL-CURRENT: 717/11; 707/203

ABSTRACT:

A distributed client-server application is modified while running. The application is not stopped so that updating of objects is transparent. A meta server catalogs all object classes for both the server and the clients. Modifications are specified by a run-time update tool and converted to change commands. The meta server receives the change commands and updates the structure of an application database. Object class definitions are read from the meta server and modified by the meta server to access the new structure of the application database. The modified object-class definitions are written back to persistent storage for the meta server, and compiled and linked to form new object classes. An object adaptor receives a list of modified object classes from the meta server and notifies all server and client caches of the object classes on the list. The obsolete objects are invalidated by the caches and new objects are created using the most up-to-date class definitions. New references to the objects are sent to the new objects, although the old objects continue to process existing references until their reference count reaches zero and they are deleted. Old and new objects co-exist for a period of time during the run-time update.

16 Claims, 14 Drawing figures Exemplary Claim Number: 1

Number of Drawing Sheets: 11

Full	Title	Citation	Front	Review	Classification	Date	Reference	Claims	K/M/C	Draw Desc	Image
------	-------	----------	-------	--------	----------------	------	-----------	--------	-------	-----------	-------

[Generate Collection](#)

Terms	Documents
(object-oriented OR (object ADJ oriented))AND (717/??ccls. OR 717/?.ccls.) AND (workflow OR work ADJ flow) AND RULES	8

[Display](#)☐

Documents, starting with Document:

[Display Format:](#)[Change Format](#)